
Cahier des charges : Conception d'Outil Décisionnel

THIBAUT AUFRERE | SAMUEL BOUSSAC |
HUGOLIN CORDEBOIS | KILIAN GIRAUD |

BUT SCIENCES DES DONNÉES | OPTION VCOD

Table des matières

1	Informations Générales	3
1.1	Contexte et Objectifs du Projet	3
1.1.1	Contexte	3
1.1.2	Objectifs du projet	3
1.1.3	Périmètre du projet	4
1.2	Enjeux et Contraintes	4
1.2.1	Enjeux	4
1.2.2	Contraintes	5
2	Description de l'Existant	5
2.1	Infrastructure et Architecture actuelle	5
2.1.1	Serveurs	5
2.1.2	Base de données	5
2.1.3	Infrastructure réseau	6
2.1.4	Outils et technologies en place	6
2.2	Description des Données Existantes	6
2.2.1	Sources de données	6
2.2.2	Qualité des données	7
2.2.3	MCD des données existantes	7
3	Description des Besoins	7
3.1	Besoins Fonctionnels	7
3.1.1	Fonctionnalités Clés du Datawarehouse	7
3.2	Besoins Techniques	9
3.2.1	Compatibilité	9
3.2.2	Performance	9
3.2.3	Sécurité	9
3.2.4	Accessibilité	9
4	Conception de la Nouvelle Solution	10
4.1	MCD du Datawarehouse	10
4.1.1	Modèle Conceptuel des Données (MCD) du Datawarehouse . .	10
4.2	Description des Tables	11
4.2.1	Tables de faits	11
4.2.2	Tables de dimensions	11
5	Spécifications Techniques	13
5.1	Création et Scripts SQL des Tables et Vues	13
5.1.1	Scripts SQL de création des vues (sur les données existantes) .	13
5.1.2	Scripts SQL de création des tables du nouveau Datawarehouse	15
5.2	Technologies et Outils	17

5.2.1	Langages de programmation	17
5.2.2	Outils de visualisation de données	17
5.2.3	Environnement de développement	17
6	Plan de Développement	17
6.1	Phases du projet	17
6.1.1	Phase 1 : Analyse et conception	17
6.1.2	Phase 2 : Développement	18
6.1.3	Phase 3 : Tests	18
6.1.4	Phase 4 : Mise en production	18
6.2	Livrables attendus	18
7	Organisation et Responsabilités	19
7.1	Equipe du Projet	19
8	Contraintes	20
8.1	Contraintes de Planning	20
8.2	Contraintes Techniques	20
9	Conditions d'Acceptation	20
9.1	Critères de validation	20
9.2	Recette finale	21
9.2.1	Recette technique	21
9.2.2	Recette fonctionnelle	21
10	Suivi et Maintenance	22
10.1	Plan de maintenance corrective, évolutive et préventive	22
10.2	Plan de suivi	22

1 Informations Générales

1.1 Contexte et Objectifs du Projet

1.1.1 Contexte

Mig Corporation est une entreprise de vente en ligne spécialisée dans la commercialisation de produits variés, destinés à une clientèle diversifiée comprenant des particuliers et des professionnels. L'activité de l'entreprise repose sur un processus bien défini incluant la gestion des commandes, la relation avec les fournisseurs, l'expédition des produits et le suivi des livraisons. Actuellement, Mig Corporation utilise une base de données opérationnelle pour conserver des informations relatives à ses clients, produits, commandes et livraisons. Toutefois, cette base est essentiellement orientée vers des besoins opérationnels et ne permet pas une analyse stratégique approfondie.

Avec la croissance rapide de l'entreprise, plusieurs problématiques sont apparues. Les équipes rencontrent des difficultés à extraire les informations essentielles pour les analyses métiers. Le traitement des données, actuellement réalisé à partir des bases transactionnelles, se révèle inefficace. Cette situation entraîne des pertes de temps considérables et un manque de visibilité sur les indicateurs clés de performance.

Ces limitations ont un impact direct sur deux processus stratégiques majeurs. Le premier concerne le suivi et l'analyse des ventes : le service financier peine à analyser les chiffres d'affaires, les remises accordées ou encore la performance des produits. Le second processus concerne la gestion et l'optimisation des livraisons, où des problèmes de suivi des délais, d'efficacité des transporteurs et de satisfaction client freinent les performances logistiques.

Pour résoudre ces difficultés, il devient essentiel de mettre en place un entrepôt de données (datawarehouse) centralisé, qui offrira des capacités d'analyse multidimensionnelle et soutiendra une prise de décision mieux informée.

1.1.2 Objectifs du projet

Objectif général L'objectif général du projet est de concevoir et implémenter un entrepôt de données centralisé. Cet outil permettra de regrouper les données opérationnelles et d'offrir des outils analytiques performants pour améliorer la prise de décision stratégique.

Les objectifs spécifiques sont les suivants :

- Améliorer le suivi et l'analyse des ventes en fournissant des outils permettant d'évaluer la rentabilité de l'entreprise, d'optimiser les politiques tarifaires et commerciales, et de procéder à une analyse approfondie des performances des produits et services.
- Optimiser la gestion des livraisons en améliorant la visibilité sur les flux logistiques, en renforçant l'efficacité des transporteurs et en élevant la satisfaction client grâce à un suivi précis des indicateurs clés.

1.1.3 Périmètre du projet

L'entrepôt de données centralisé intégrera les informations suivantes issues du système transactionnel actuel :

- **Données relatives aux clients, commandes et détails des commandes :** Ces informations permettront une analyse approfondie des ventes, en identifiant les tendances par client, produit ou région.
- **Données sur les transporteurs et les informations de livraison :** Elles seront exploitées pour suivre les expéditions et optimiser les performances logistiques.
- **Données sur les produits et les fournisseurs :** Ces données serviront à évaluer les performances des produits et à améliorer les relations avec les fournisseurs grâce à une analyse de leur efficacité et de leur fiabilité.

En regroupant ces informations, Mig Corporation pourra optimiser ses performances commerciales et logistiques tout en renforçant la satisfaction client.

1.2 Enjeux et Contraintes

1.2.1 Enjeux

Ce projet revêt une importance stratégique pour Mig Corporation, car il vise à résoudre des problèmes clés qui freinent actuellement les performances de l'entreprise. Les principaux enjeux sont les suivants :

- **Gain de temps** : La centralisation des données et les outils d'analyse permettront de réduire les délais liés à l'extraction et au traitement des données.
- **Optimisation des ressources** : Une meilleure exploitation des données existantes permettra d'optimiser les opérations internes et de soutenir les décisions stratégiques.
- **Amélioration de la compétitivité** : En renforçant les performances commerciales et logistiques, Mig Corporation pourra se positionner comme leader sur son marché.

1.2.2 Contraintes

Malgré ces avantages, le projet devra faire face à plusieurs contraintes :

- **Contraintes budgétaires** : Le projet devra être réalisé dans le cadre du budget alloué.
- **Contraintes techniques** : L'entrepôt devra être compatible avec le système d'information existant et capable de traiter de grands volumes de données.
- **Contraintes humaines** : Le projet nécessitera la mobilisation des équipes internes et la formation des utilisateurs finaux.
- **Contraintes légales** : Le traitement des données devra être conforme aux réglementations, notamment le RGPD.
- **Contraintes de délais** : Le projet devra respecter les échéances prévues pour une mise en œuvre rapide et efficace.

2 Description de l'Existant

2.1 Infrastructure et Architecture actuelle

2.1.1 Serveurs

L'entreprise utilise actuellement une base de données relationnelle pour stocker les informations relatives à ses commandes, clients, produits, employés et autres entités métiers. La localisation et les caractéristiques des serveurs ne sont pas précisées dans le sujet, mais nous supposons qu'il s'agit d'un système centralisé sur un serveur unique ou un petit cluster.

2.1.2 Base de données

La base de données actuelle est structurée selon un modèle relationnel, avec plusieurs tables interdépendantes :

- **Categorie** : Informations sur les catégories de produits.
- **Client** : Détails sur les clients de l'entreprise.
- **Commande** : Historique des commandes effectuées par les clients.
- **DetailCommande** : Informations détaillées sur chaque ligne de commande.
- **Employe** : Détails sur les employés impliqués dans les processus commerciaux.
- **Fournisseur** : Informations sur les fournisseurs des produits.
- **Transporteur** : Données sur les transporteurs responsables des livraisons.
- **Produit** : Catalogue des produits avec leurs attributs et quantités en stock.

Ces tables sont reliées entre elles par des clés primaires et étrangères, permettant une gestion cohérente des données transactionnelles.

2.1.3 Infrastructure réseau

L'infrastructure réseau n'est pas explicitement détaillée dans le sujet. Nous supposons qu'il existe une connectivité basique entre les utilisateurs finaux (employés, analystes) et le serveur de base de données via un réseau local ou une solution cloud.

2.1.4 Outils et technologies en place

Les systèmes et outils utilisés pour l'analyse ou le reporting ne sont pas précisés. Cependant, la structure relationnelle existante semble optimisée pour des opérations transactionnelles plutôt que pour des analyses multidimensionnelles.

2.2 Description des Données Existantes

2.2.1 Sources de données

Les données proviennent des tables relationnelles décrites dans le sujet, qui regroupent :

- **Données clients** : Sociétés, contacts, adresses, pays.
- **Données commandes** : Dates, clients, transporteurs, détails des produits commandés.
- **Données produits** : Fournisseurs, quantités en stock, catégories.
- **Données livraisons** : Destinations, coûts, transporteurs.

2.2.2 Qualité des données

- Le problèmes de qualité potentiels incluent :
 - **Données manquantes** : Certaines colonnes, comme les numéros de téléphone ou les codes postaux, peuvent être incomplètes.
 - **Doublons** : Existence possible de clients ou produits dupliqués.
 - **Incohérences** : Informations déphasées entre les tables (par exemple, commandes référant à des produits supprimés).

2.2.3 MCD des données existantes

Le Modèle Conceptuel des Données (MCD) des données existantes est conçu pour optimiser les opérations transactionnelles. Voici un résumé des entités et relations principales :

- Client (1,n)—(0,n) Commande
- Commande (1,n)—(1,n) DetailCommande
- Produit (1,n)—(1,n) DetailCommande
- Produit (0,n)—(1,1) Categorie
- Commande (1,1)—(0,n) Employe
- Commande (1,1)—(0,n) Transporteur
- Produit (1,n)—(1,1) Fournisseur

Ce MCD met en évidence la structure relationnelle existante et les dépendances clés entre les entités. Cependant, cette structure n'est pas adaptée pour les analyses multidimensionnelles, ce qui justifie la création d'un entrepôt de données.

3 Description des Besoins

3.1 Besoins Fonctionnels

3.1.1 Fonctionnalités Clés du Datawarehouse

- **Extraction, Transformation et Chargement (ETL)**
 - Mise en place d'un processus ETL automatisé pour extraire les données des systèmes sources, les transformer selon les règles d'intégration et les charger dans le Datawarehouse.
 - Intégration des données en provenance des bases de données opérationnelles (à partir des tables Client, Commande, DetailCommande, Produit, etc.).
 - Nettoyage des données (gestion des doublons, correction des erreurs de saisie, standardisation des formats, etc.).

- **Gestion des Requêtes Analytiques**
 - Prise en charge des requêtes complexes pour l'analyse des performances des ventes et des livraisons.
 - Possibilité de requêtes multi-dimensionnelles (OLAP) pour analyser les ventes et les livraisons selon les dimensions telles que le temps, la géographie, les clients, les produits, etc.
 - Accès à des tableaux de bord et rapports préétablis.
- **Gestion des Vues et des Agrégations**
 - Préparation de vues pré-agrégées pour améliorer la vitesse de réponse des requêtes analytiques.
 - Possibilité de créer des indicateurs clés de performance (KPI) spécifiques aux services de la finance et de la logistique.
- **Suivi des Mises à Jour**
 - Gestion des historiques des commandes et des livraisons (suivi des modifications des données à travers le temps).
 - Fonctionnalité de traçabilité des opérations de mise à jour des données.
- **Accès à l'Information**
 - Mise à disposition d'une interface utilisateur intuitive permettant la création de requêtes personnalisées.
 - Fonctionnalités de visualisation de données par le biais de tableaux de bord, de graphiques et de rapports.
- **Maintenance et Administration**
 - Outils de surveillance et de gestion de la performance du Datawarehouse.
 - Outils de gestion des sauvegardes et de la récupération des données en cas de sinistre.

3.1.2. Utilisateurs Cibles

- **Utilisateurs Finaux**
 - Service Finance : analyse des performances de vente (chiffre d'affaires, marges, etc.).
 - Service Logistique : suivi des délais de livraison et analyse de l'efficacité logistique.
- **Utilisateurs Techniques**
 - Administrateurs de bases de données : responsables de la gestion, de la maintenance et de la sécurité du Datawarehouse.
 - Analystes et développeurs BI : chargés de développer des requêtes, des rapports et des tableaux de bord.
- **Autres Acteurs**
 - Décideurs stratégiques (direction) : accès aux rapports synthétiques et à la visualisation des performances globales.

3.2 Besoins Techniques

3.2.1 Compatibilité

- **Systèmes d'Exploitation (OS)** : Le Datawarehouse doit être compatible avec les principaux systèmes d'exploitation (Windows Server, Linux/Unix).
- **Systèmes de Gestion de Bases de Données (SGBD)** : Le Datawarehouse devra être compatible avec les SGBD relationnels (MySQL, PostgreSQL, SQL Server, Oracle) et les bases NoSQL (MongoDB, Cassandra) pour permettre une intégration hybride.
- **Outils ETL et BI** : Compatibilité avec les outils ETL tels que Talend, Pentaho ou SSIS pour l'automatisation des flux ETL.

3.2.2 Performance

- **Gestion du Volume de Données**
- Capacité à gérer un volume de données en constante augmentation.
- Support de la parallélisation des processus ETL et des requêtes analytiques.
- **Scalabilité** : Possibilité de mise à l'échelle horizontale et verticale.
- **Temps de Réponse** : Optimisation des performances pour garantir des temps de réponse courts.

3.2.3 Sécurité

- **Gestion des Accès** : Mise en place d'un système d'authentification et de gestion des droits d'accès par rôle.
- **Confidentialité des Données** : Chiffrement des données sensibles.
- **Surveillance de la Sécurité** : Mise en place de journaux d'audit.

3.2.4 Accessibilité

- **Interface Utilisateur** : Interface web intuitive et responsive accessible depuis les navigateurs courants.
- **Accès Multi-Plateforme** : Accès au Datawarehouse depuis des ordinateurs, tablettes et smartphones.
- **Accès 24/7** : Disponibilité continue du Datawarehouse.

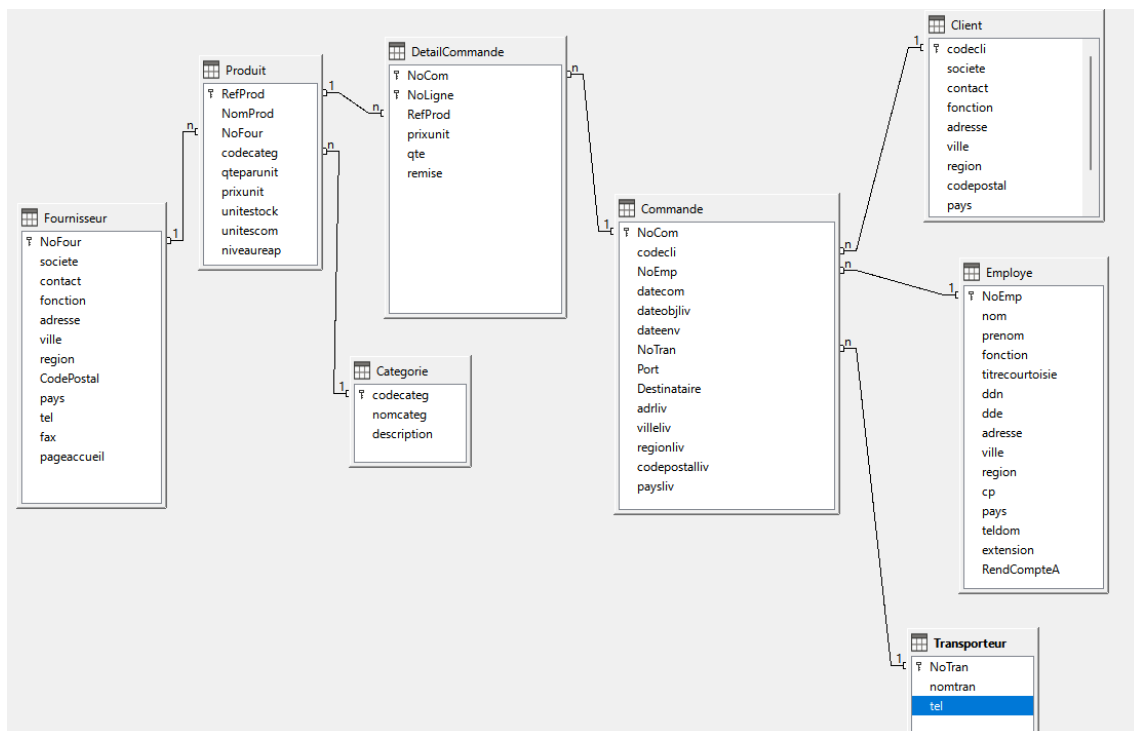
4 Conception de la Nouvelle Solution

4.1 MCD du Datawarehouse

4.1.1 Modèle Conceptuel des Données (MCD) du Datawarehouse

- Schéma MCD du nouveau Datawarehouse.
- Identification des tables de faits, des dimensions et des relations entre elles.

Le Modèle Conceptuel des Données (ou MCD) du Datawarehouse est une représentation des données clés qui seront intégrées, organisées et exploitées dans le cadre de futur analyses décisionnelles.



Le MCD présenté ci-dessus illustre les principales entités du Datawarehouse, notamment les clients, les produits, les commandes et leurs relations. Ce modèle met en évidence les dépendances logiques entre les données qui alimenteront le système d'analyse.

4.2 Description des Tables

4.2.1 Tables de faits

Les tables de faits contiennent les mesures quantitatives qui seront analysées. Elles représentent les événements ou transactions d'intérêt dans le Datawarehouse. Dans ce cas, la table principale est DetailCommandes.

Cette table capture les données transactionnelles liées aux commandes effectuées par les clients. Elle inclut des indicateurs mesurables qui serviront aux analyses, comme les quantités commandées, les remises appliquées et les prix unitaires.

Cette table est au cœur des analyses, car elle connecte l'ensemble des acteurs de notre environnement.

Colonne	Type	Desc
NoCom	Clé étrangère	Référence unique liée à une commande spécifique
NoLigne	Clé primaire	Numéro de ligne pour distinguer le produit dans une commande
RefProd	Clé étrangère	Référence au produit commandé
prixunit	Float	Prix unitaire du produit au moment de la commande
qte	Int	Quantité commandée pour ce produit
remise	Float	Pourcentage de remise appliqué sur cette ligne de commande

A l'aide de notre base nous pouvons alors obtenir plusieurs mesures clés comme :

- La quantité totale commandée ou alors le montant total par ligne de commande ou bien la remise totale appliquée. Ces informations ne représentent qu'une petite partie de ce qu'il sera possible d'obtenir.

4.2.2 Tables de dimensions

Les tables de dimensions offrent des informations contextuelles sur les données stockées dans les tables de faits. Elles permettent d'analyser les faits selon différents angles, comme celui des produits, de la clientèle ou encore au niveau des stocks.

La table **PRODUIT** contient les informations sur les produits disponibles à la vente. Chaque produit est identifié par une référence unique et est enrichi par des données comme son nom, son prix, sa catégorie et son niveau de stock.

Colonne	Type	Desc
RefProd	Clé primaire	Référence unique du produit
NomProd	Varchar	Nom du produit
NoFour	Clé étrangère	Fournisseur du produit
codecateg	Clé étrangère	Catégorie du produit
qteparunit	Int	Quantité commandée pour ce produit
prixunit	Float	Prix unitaire standard
unitestock	Int	Quantité disponible en stock
niveaureap	Int	Niveau de réapprovisionnement minimal

La table **CLIENT** regroupe les informations sur les clients ayant passé des commandes. Chaque client est identifié par un code unique (codecli) et décrit par des attributs comme son nom, son adresse et sa localisation.

Colonne	Type	Desc
codecli	Clé primaire	Code unique du client
societe	Varchar	Nom de l'entreprise ou du client
contact	Varchar	Nom de la personne à contacter
fonction	Varchar	Fonction de la personne à contacter
adresse	Varchar	Adresse complète du client
ville	Varchar	Ville du client
region	Varchar	Région administrative
codepostal	Varchar	Code postal
pays	Varchar	Pays d'origine

La table **FOURNISSEUR** décrit les entités qui fournissent les produits. Elle est utile pour suivre les relations fournisseurs-produits et pour analyser les performances des fournisseurs.

Colonne	Type	Desc
NoFour	Clé primaire	Identifiant unique du fournisseur
societe	Varchar	Nom de l'entreprise ou du client
contact	Varchar	Nom de la personne à contacter
fonction	Varchar	Fonction de la personne à contacter
adresse	Varchar	Adresse complète du client
ville	Varchar	Ville du client
region	Varchar	Région administrative
codepostal	Varchar	Code postal
pays	Varchar	Pays d'origine

La table **EMPLOYE** contient les informations sur les employés impliqués dans la gestion des commandes (par exemple, commerciaux ou responsables logistiques).

Colonne	Type	Desc
NoFour	Clé primaire	Identifiant unique du fournisseur
societe	Varchar	Nom de l'entreprise ou du client
contact	Varchar	Nom de la personne à contacter
fonction	Varchar	Fonction de la personne à contacter
titrecourtoisie	Varchar	Titre de civilité
adresse	Varchar	Adresse complète du client
ddn	Date	Date de naissance de l'employe
ville	Varchar	Ville du client
region	Varchar	Région administrative
codepostal	Varchar	Code postal
pays	Varchar	Pays d'origine

5 Spécifications Techniques

5.1 Création et Scripts SQL des Tables et Vues

5.1.1 Scripts SQL de création des vues (sur les données existantes)

Pour permettre une analyse des processus métiers existants sans perturber les données transactionnelles, des vues seront créées sur les tables relationnelles actuelles. Voici quelques exemples de scripts pour créer ces vues :

1. Vue des commandes avec détails des produits :

```
1 CREATE VIEW vue_commandes_details AS
2 SELECT
3     c.NoCom,
4     c.datecom,
5     c.dateenv,
6     d.RefProd,
7     p.nomprod,
8     d.qte,
9     d.prixunit,
10    d.remise
11 FROM
12     Commande c
13 JOIN
14     DetailCommande d ON c.NoCom = d.NoCom
15 JOIN
16     Produit p ON d.RefProd = p.RefProd;
```

2. Vue des clients et commandes associées :

```
1 CREATE VIEW vue_clients_commandes AS
2 SELECT
3     cli.codecli,
4     cli.societe,
5     cli.pays,
6     com.NoCom,
7     com.datecom,
8     com.dateobjliv
9 FROM
10     Client cli
11 JOIN
12     Commande com ON cli.codecli = com.codecli;
```

3. Vue des produits par catégorie :

```
1 CREATE VIEW vue_produits_categorie AS
2 SELECT
3     p.RefProd,
4     p.nomprod,
5     c.codecateg,
6     c.nomcateg,
7     c.description
8 FROM
9     Produit p
10 JOIN
11     Categorie c ON p.codecateg = c.codecateg;
```

5.1.2 Scripts SQL de création des tables du nouveau Datawarehouse

Le Datawarehouse sera conçu en suivant une architecture en étoile, avec une table de faits principale et plusieurs tables de dimensions.

1. Table de faits :

```
1 CREATE TABLE Faits_Ventes (
2     id_fact INT PRIMARY KEY AUTO_INCREMENT,
3     NoCom INT,
4     RefProd INT,
5     datecom DATE,
6     montant_total DECIMAL(10,2),
7     quantite INT,
8     remise DECIMAL(5,2),
9     FOREIGN KEY (NoCom) REFERENCES Commande(NoCom),
10    FOREIGN KEY (RefProd) REFERENCES Produit(RefProd)
11 );
```


2. Tables de dimensions | Dimension Client :

```
1 CREATE TABLE Dim_Client (  
2     id_client INT PRIMARY KEY AUTO_INCREMENT,  
3     codecli INT,  
4     societe VARCHAR(255),  
5     pays VARCHAR(100),  
6     ville VARCHAR(100),  
7     FOREIGN KEY (codecli) REFERENCES Client(codecli)  
8 );
```

2. Tables de dimensions | Dimension Produit :

```
1 CREATE TABLE Dim_Produit (  
2     id_produit INT PRIMARY KEY AUTO_INCREMENT,  
3     RefProd INT,  
4     nomprod VARCHAR(255),  
5     codecateg INT,  
6     FOREIGN KEY (RefProd) REFERENCES Produit(RefProd),  
7     FOREIGN KEY (codecateg) REFERENCES Categorie(codecateg)  
8 );
```

2. Tables de dimensions | Dimension Temps :

```
1 CREATE TABLE Dim_Temps (  
2     id_temps INT PRIMARY KEY AUTO_INCREMENT,  
3     date_com DATE,  
4     annee INT,  
5     mois INT,  
6     jour INT  
7 );
```

5.2 Technologies et Outils

5.2.1 Langages de programmation

- **SQL** : Utilisé pour la création des tables, des vues et des requêtes analytiques.
- **Python / Talend** : Pour les étapes d'ETL (Extraction, Transformation et Chargement) et l'automatisation des flux de données.

5.2.2 Outils de visualisation de données

- **Power BI/Tableau** : Visualisation et analyse des données agrégées.
- **MySQL/PostgreSQL** : Gestion des bases de données relationnelles et création des vues.

5.2.3 Environnement de développement

- **IDE** : VS Code ou PyCharm pour éditer les scripts SQL et Python.
- **Serveurs** : AWS RDS ou un serveur local pour déployer la base de données et le Datawarehouse.

Ces spécifications fournissent une base solide pour le développement du projet tout en assurant une compatibilité avec les technologies existantes et une facilité de maintenance future.

6 Plan de Développement

6.1 Phases du projet

6.1.1 Phase 1 : Analyse et conception

- Recueil des besoins auprès des parties prenantes (finance, logistique, DSI).
- Réalisation et validation des modèles conceptuels des données (MCD) et des modèles logiques des données (MLD).
- Validation des spécifications fonctionnelles et techniques du Datawarehouse.

6.1.2 Phase 2 : Développement

- Conception des bases de données physiques et des scripts de création des tables SQL.
- Mise en place de l'ETL (Extraction, Transformation, Chargement) pour intégrer les données opérationnelles dans le Datawarehouse.
- Développement des scripts de création des vues SQL.
- Développement des dimensions et des tables de faits spécifiques aux processus de gestion des ventes et des livraisons.

6.1.3 Phase 3 : Tests

- Réalisation des tests unitaires sur chaque composant (tables, vues, ETL, agrégations, etc.).
- Réalisation des tests d'intégration pour vérifier la cohésion des différents composants.
- Tests de performance pour évaluer la vitesse des requêtes et la capacité à gérer de grands volumes de données.

6.1.4 Phase 4 : Mise en production

- Livraison finale du Datawarehouse aux services de la finance et de la logistique.
- Formation des utilisateurs finaux, analystes et administrateurs de base de données.
- Mise en place des outils de supervision et de maintenance (sauvegardes, restauration, surveillance de la performance).

6.2 Livrables attendus

- **MCD du Datawarehouse** : Le Modèle Conceptuel des Données (MCD) qui représente les entités, les relations et les cardinalités.
- **Scripts de création SQL** : Scripts de création des tables (faits et dimensions) et des vues (vues d'analyse et vues pré-agrégées).
- **Rapport de tests** : Rapport de validation des tests unitaires, des tests d'intégration et des tests de performance, présentant les éventuelles anomalies corrigées et les performances finales atteintes.

7 Organisation et Responsabilités

7.1 Equipe du Projet

L'équipe du projet se compose de plusieurs acteurs, chacun ayant un rôle précis et des responsabilités bien définies pour assurer le bon déroulement des différentes phases. Le **chef de projet** est responsable de la coordination globale du projet. Il veille au respect des objectifs fixés, des délais et des contraintes budgétaires. Il agit comme un point de contact central entre les différentes parties prenantes, s'assure de la communication efficace entre les membres de l'équipe et supervise l'ensemble des activités afin de garantir leur alignement avec les attentes définies.

L'**analyste fonctionnel** joue un rôle clé dans la compréhension et la formalisation des besoins des utilisateurs. Il analyse les processus métier existants, identifie les exigences fonctionnelles et les traduit en spécifications claires et précises. Il participe également à la validation des livrables pour s'assurer que les besoins des utilisateurs finaux sont pleinement satisfaits.

Le **développeur SQL** est chargé de concevoir et de rédiger les scripts nécessaires à la création des bases de données et des vues. Il optimise les requêtes pour garantir des performances élevées et travaille en étroite collaboration avec l'analyste fonctionnel pour s'assurer que les structures de données répondent aux besoins opérationnels.

Le **spécialiste ETL** intervient dans la phase de transformation des données. Il conçoit, développe et maintient les processus d'extraction, de transformation et de chargement des données (ETL). Ce rôle implique la gestion des flux de données entre les systèmes sources et le Datawarehouse, en assurant la qualité et la cohérence des données tout au long du processus.

Enfin, l'**administrateur de bases de données (DBA)** est responsable de la gestion et de la maintenance des bases de données. Il s'assure de leur disponibilité, de leur sécurité et de leur performance. Il supervise également les sauvegardes, la récupération des données et la gestion des accès afin de garantir la protection et la continuité des services.

8 Contraintes

8.1 Contraintes de Planning

Le projet est soumis à des contraintes temporelles strictes, définies par une durée globale allant de la date de début en janvier 2025 jusqu'à la date de fin en mars 2025, afin de respecter les délais imposés par les parties prenantes. Chaque phase du projet est jalonnée par des étapes clés avec des dates de livraison intermédiaires et finales. Par exemple, la conception du Modèle Conceptuel des Données (MCD) et la validation des besoins doivent être réalisées dans les deux premières semaines du projet, tandis que les scripts SQL et les tests unitaires devront être livrés dans un délai de trois mois. Enfin, la mise en production du Datawarehouse, accompagnée de la recette finale, est prévue pour la dernière semaine du calendrier établi. Le respect de ce planning est essentiel pour garantir l'atteinte des objectifs tout en minimisant les impacts opérationnels sur les utilisateurs finaux.

8.2 Contraintes Techniques

Le projet doit également faire face à des contraintes techniques liées à l'intégration et au bon fonctionnement de la solution avec les systèmes existants. En premier lieu, la compatibilité avec les infrastructures actuelles, notamment les systèmes d'exploitation et les bases de données relationnelles en place, constitue une exigence prioritaire. La solution devra être interopérable avec les outils en usage dans l'entreprise, tels que les plateformes d'analyse décisionnelle ou les logiciels d'extraction de données.

Par ailleurs, la sécurité et la confidentialité des données sont des enjeux majeurs. Le Datawarehouse devra répondre aux normes en vigueur, notamment en matière de protection des données sensibles, comme le RGPD. Cela inclut le chiffrement des données en transit et au repos, la gestion rigoureuse des accès par authentification et autorisation, ainsi que la mise en place de journaux d'audit pour assurer une traçabilité complète des opérations. Toute vulnérabilité potentielle devra être identifiée et résolue avant la mise en production pour prévenir les risques liés à des violations de données ou des cyberattaques.

9 Conditions d'Acceptation

9.1 Critères de validation

Pour garantir que le projet répond aux attentes et exigences définies, les critères suivants seront appliqués :

- **Validation des MCD et MLD :** Les Modèles Conceptuels et Logiques des Données seront validés pour assurer leur cohérence et leur adéquation aux besoins.
- **Validation des scripts SQL :** Les scripts de création des tables et des vues seront testés pour garantir leur bon fonctionnement.
- **Bon fonctionnement de l'ETL :** L'extraction, la transformation et le chargement des données devront se dérouler sans erreur, avec des performances acceptables.
- **Respect des délais :** Les livrables devront être fournis dans les délais fixés sans compromettre la qualité.
- **Conformité aux besoins exprimés :** Le Datawarehouse devra répondre aux besoins fonctionnels et techniques spécifiés dans le cahier des charges.

9.2 Recette finale

La recette finale se déroulera en deux phases principales :

9.2.1 Recette technique

- **Validation des performances :** Les temps de réponse devront être inférieurs à X secondes pour les requêtes standards.
- **Compatibilité :** L'entrepôt de données devra fonctionner avec les systèmes existants sans conflit.
- **Sécurité :** Les mécanismes de chiffrement et de contrôle d'accès seront testés pour garantir la confidentialité et l'intégrité des données.

9.2.2 Recette fonctionnelle

- **Validation de la conformité :** Les tableaux de bord et les rapports générés devront être conformes aux attentes des services des finances et de la logistique.
- **Tests utilisateurs :** Les utilisateurs finaux effectueront des tests pour s'assurer que le Datawarehouse répond à leurs besoins opérationnels.

Les livrables finaux seront considérés comme acceptés une fois que toutes les validations auront été complétées avec succès et documentées dans un rapport de recette.

10 Suivi et Maintenance

10.1 Plan de maintenance corrective, évolutive et préventive

Maintenant corrective :

- Identification et correction des bugs dans les scripts SQL ou les processus ETL.
- Surveillance des journaux d'erreurs pour détecter les anomalies.

Maintenance évolutive :

- Ajout de nouvelles dimensions ou indicateurs dans le Datawarehouse pour répondre à des besoins futurs.
- Mise à jour des vues et des tables pour intégrer de nouvelles sources de données ou modifications dans les processus métiers.

Maintenance préventive :

- Optimisation régulière des requêtes SQL et des performances des bases de données.
- Révision des processus ETL pour éviter les ralentissements ou erreurs dus à l'augmentation des volumes de données.

10.2 Plan de suivi

Indicateurs de performance :

- Temps de réponse des requêtes analytiques.
- Taux de succès des processus ETL (chargement des données sans erreurs).
- Disponibilité du système (à surveiller via des outils comme AWS CloudWatch ou Zabbix).

Responsabilités de maintenance :

- **Administrateur de bases de données (DBA)** : Assure la continuité des opérations du Datawarehouse et optimise les performances.
- **Spécialiste ETL** : Gère les pipelines de données et résout les problèmes liés aux processus d'intégration.
- **Analyste des données** : Propose des évolutions fonctionnelles en fonction des besoins des utilisateurs.

Outils de suivi :

- **JIRA/Trello** : Pour la gestion des tickets et des demandes de maintenance.
- **Power BI/Tableau** : Pour surveiller les performances des différentes couches du Datawarehouse.
- **GitHub/GitLab** : Pour la gestion des versions des scripts et des configurations du projet.

Ce plan de maintenance garantit la pérennité, l'efficacité et l'évolutivité du Datawarehouse tout en minimisant les interruptions de service.